JSP - SENDING EMAIL

To send an email using a JSP is simple enough but to start with you should have **JavaMail API** and **Java Activation Framework (JAF)** installed on your machine.

- You can download latest version of <u>JavaMail (Version 1.2)</u> from Java's standard website.

- You can download latest version of JavaBeans Activation Framework <u>JAF (Version 1.0.2)</u> from Java's standard website.

Download and unzip these files, in the newly created top level directories you will find a number of jar files for both the applications. You need to add **mail.jar** and **activation.jar** files in your CLASSPATH.

Send a Simple Email:

Here is an example to send a simple email from your machine. Here it is assumed that your **localhost** is connected to the internet and capable enough to send an email. Same time make sure all the jar files from Java Email API package and JAF package ara available in CLASSPATH.

```
<%@ page import="java.io.*,java.util.*,javax.mail.*"%>
<%@ page import="javax.mail.internet.*,javax.activation.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%
   String result;
   // Recipient's email ID needs to be mentioned.
   String to = "abcd@gmail.com";

   // Sender's email ID needs to be mentioned
   String from = "mcmohd@gmail.com";

   // Assuming you are sending email from localhost
   String host = "localhost";

   // Get system properties object
   Properties properties = System.getProperties();

   // Setup mail server
   properties.setProperty("mail.smtp.host", host);

   // Get the default Session object.
   Session mailSession = Session.getDefaultInstance(properties);

   try{
      // Create a default MimeMessage object.
      MimeMessage message = new MimeMessage(mailSession);
      // Set From: header field of the header.
      message.setFrom(new InternetAddress(from));
      // Set To: header field of the header.
      message.addRecipient(Message.RecipientType.TO,
                              new InternetAddress(to));
      // Set Subject: header field
      message.setSubject("This is the Subject Line!");
      // Now set the actual message
      message.setText("This is actual message");
      // Send message
      Transport.send(message);
      result = "Sent message successfully....";
   }catch (MessagingException mex) {
      mex.printStackTrace();
      result = "Error: unable to send message....";
   }
%>
```

```
<html>
<head>
<title>Send Email using JSP</title>
</head>
<body>
<center>
<h1>Send Email using JSP</h1>
</center>
<p align="center">
<%
    out.println("Result: " + result + "\n");
%>
</p>
</body>
</html>
```

Now let us put above code in SendEmail.jsp file and call this JSP using URL http://localhost:8080/SendEmail.jsp which would send an email to given email ID *abcd@gmail.com* and would display following response:

SEND EMAIL USING JSP

Result: Sent message successfully....

If you want to send an email to multiple recipients then following methods would be used to specify multiple email IDs:

```
void addRecipients(Message.RecipientType type,
                   Address[] addresses)
throws MessagingException
```

Here is the description of the parameters:

- **type:** This would be set to TO, CC or BCC. Here CC represents Carbon Copy and BCC represents Black Carbon Copy. Example *Message.RecipientType.TO*

- **addresses:** This is the array of email ID. You would need to use InternetAddress() method while specifying email IDs

Send an HTML Email:

Here is an example to send an HTML email from your machine. Here it is assumed that your **localhost** is connected to the internet and capable enough to send an email. Same time make sure all the jar files from Java Email API package and JAF package ara available in CLASSPATH.

This example is very similar to previous one, except here we are using setContent() method to set content whose second argument is "text/html" to specify that the HTML content is included in the message.

Using this example, you can send as big as HTML content you like.

```
<%@ page import="java.io.*,java.util.*,javax.mail.*"%>
<%@ page import="javax.mail.internet.*,javax.activation.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%
    String result;
    // Recipient's email ID needs to be mentioned.
    String to = "abcd@gmail.com";

    // Sender's email ID needs to be mentioned
    String from = "mcmohd@gmail.com";
```

```
   // Assuming you are sending email from localhost
   String host = "localhost";

   // Get system properties object
   Properties properties = System.getProperties();

   // Setup mail server
   properties.setProperty("mail.smtp.host", host);

   // Get the default Session object.
   Session mailSession = Session.getDefaultInstance(properties);

   try{
      // Create a default MimeMessage object.
      MimeMessage message = new MimeMessage(mailSession);
      // Set From: header field of the header.
      message.setFrom(new InternetAddress(from));
      // Set To: header field of the header.
      message.addRecipient(Message.RecipientType.TO,
                            new InternetAddress(to));
      // Set Subject: header field
      message.setSubject("This is the Subject Line!");

      // Send the actual HTML message, as big as you like
      message.setContent("<h1>This is actual message</h1>",
                         "text/html" );
      // Send message
      Transport.send(message);
      result = "Sent message successfully....";
   }catch (MessagingException mex) {
      mex.printStackTrace();
      result = "Error: unable to send message....";
   }
%>
<html>
<head>
<title>Send HTML Email using JSP</title>
</head>
<body>
<center>
<h1>Send Email using JSP</h1>
</center>
<p align="center">
<%
   out.println("Result: " + result + "\n");
%>
</p>
</body>
</html>
```

Now try to use above JSP to send HTML message on a given email ID.

Send Attachment in Email:

Here is an example to send an email with attachment from your machine:

```
<%@ page import="java.io.*,java.util.*,javax.mail.*"%>
<%@ page import="javax.mail.internet.*,javax.activation.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%
   String result;
   // Recipient's email ID needs to be mentioned.
   String to = "abcd@gmail.com";

   // Sender's email ID needs to be mentioned
   String from = "mcmohd@gmail.com";

   // Assuming you are sending email from localhost
   String host = "localhost";
```

```jsp
   // Get system properties object
   Properties properties = System.getProperties();

   // Setup mail server
   properties.setProperty("mail.smtp.host", host);

   // Get the default Session object.
   Session mailSession = Session.getDefaultInstance(properties);

   try{
      // Create a default MimeMessage object.
      MimeMessage message = new MimeMessage(mailSession);

      // Set From: header field of the header.
      message.setFrom(new InternetAddress(from));

      // Set To: header field of the header.
      message.addRecipient(Message.RecipientType.TO,
                              new InternetAddress(to));

      // Set Subject: header field
      message.setSubject("This is the Subject Line!");

      // Create the message part
      BodyPart messageBodyPart = new MimeBodyPart();

      // Fill the message
      messageBodyPart.setText("This is message body");

      // Create a multipar message
      Multipart multipart = new MimeMultipart();

      // Set text message part
      multipart.addBodyPart(messageBodyPart);

      // Part two is attachment
      messageBodyPart = new MimeBodyPart();
      String filename = "file.txt";
      DataSource source = new FileDataSource(filename);
      messageBodyPart.setDataHandler(new DataHandler(source));
      messageBodyPart.setFileName(filename);
      multipart.addBodyPart(messageBodyPart);

      // Send the complete message parts
      message.setContent(multipart );

      // Send message
      Transport.send(message);
      String title = "Send Email";
      result = "Sent message successfully....";
   }catch (MessagingException mex) {
      mex.printStackTrace();
      result = "Error: unable to send message....";
   }
%>
<html>
<head>
<title>Send Attachement Email using JSP</title>
</head>
<body>
<center>
<h1>Send Attachement Email using JSP</h1>
</center>
<p align="center">
<%
   out.println("Result: " + result + "\n");
%>
</p>
</body>
</html>
```

Now try to run above JSP to send a file as an attachement along with a message on a given email ID.

User Authentication Part:

If it is required to provide user ID and Password to the email server for authentication purpose then you can set these properties as follows:

```
props.setProperty("mail.user", "myuser");
props.setProperty("mail.password", "mypwd");
```

Rest of the email sending mechanism would remain as explained above.

Using Forms to send email:

You can use HTML form to accept email parameters and then you can use **request** object to get all the information as follows:

```
String to = request.getParameter("to");
String from = request.getParameter("from");
String subject = request.getParameter("subject");
String messageText = request.getParameter("body");
```

Once you have all the information, you can use above mentioned programs to send email.